(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

PARIS

(11) N° de publication : **2 594 573**
(à n'utiliser que pour les commandes de reproduction)

(21) N° d'enregistrement national : **87 00343**

(51) Int Cl⁴ : G 06 K 19/00; G 06 F 11/08.

(12) DEMANDE DE BREVET D'INVENTION A1

(22) Date de dépôt : 14 janvier 1987.

(30) Priorité : JP, 18 février 1986, n° 33345/86.

(43) Date de la mise à disposition du public de la demande : BOPI « Brevets » n° 34 du 21 août 1987.

(60) Références à d'autres documents nationaux apparentés :

(71) Demandeur(s) : *Société dite : KABUSHIKI KAISHA TOSHIBA.* — JP.

(72) Inventeur(s) : Yasuo Iijima.

(73) Titulaire(s) :

(74) Mandataire(s) : Cabinet Beau de Loménie.

(54) Dispositif électronique portatif.

(57) Le dispositif électronique portatif de l'invention possède une mémoire de données, qui est une mémoire permanente, pour emmagasiner des données de transaction et une unité centrale de traitement qui exécute un accès en lecture et écriture de données à la mémoire. L'unité centrale de traitement produit un code de somme de contrôle ou un code ayant subi une fonction OU exclusif pour la donnée de transaction, compte la longueur de la donnée de transaction contenant le code et adjoint le code et la longueur à la donnée de transaction en vue de leur emmagasinage dans la mémoire de données. Pour la lecture de la donnée, l'unité centrale de traitement produit un code de somme de contrôle ou un code ayant subi une fonction OU exclusif pour la donnée de transaction, compte la longueur de la donnée, y compris le code, puis compare ces valeurs avec celles se trouvant dans la mémoire de données. Lorsqu'une coïncidence est trouvée, l'unité centrale de traitement décide que la donnée écrite en mémoire est valable.

FR 2 594 573 - A1

La présente invention concerne un dispositif électronique portatif, plus spécialement ce que l'on appelle une carte à circuit intégré, qui est utilisé dans un système destiné à permettre d'effectuer des opérations bancaires ou des achats et qui contient une puce de circuit intégré possédant une mémoire de données permanente et un élément de commande, par exemple une unité centrale de traitement (CPU).

Ces dernières années, on a proposé une carte à circuit intégré contenant une puce de circuit intégré qui possède une mémoire de données permanente et un élément de commande (par exemple un CPU).

Les données emmagasinées dans la mémoire de cette carte à circuit intégré sont gérées par un élément de commande que celle-ci contient ou par un lecteur-enregistreur connecté à la carte à circuit intégré.

Il a été proposé, comme application pratique de cette carte à circuit intégré, un système permettant d'effectuer des opérations bancaires ou des achats. Dans cette application, il faut pouvoir vérifier la validité des données. Lorsque l'on écrit une donnée dans la mémoire, on peut lire celle-ci afin de vérifier si elle a été correctement écrite. Toutefois, lorsqu'on lit la donnée écrite dans la mémoire, il n'existe aucun moyen de vérifier si elle a été correctement lue.

C'est un but de l'invention de proposer un dispositif électronique portatif qui peut vérifier de manière fiable la validité d'une donnée, écrite en mémoire, lorsqu'elle est lue, ce qui améliore la fiabilité de la donnée.

Pour réaliser le but ci-dessus indiqué, il est proposé un dispositif électronique portatif comprenant :

un moyen de mémoire permettant d'emmagasiner une donnée de transaction; et

une unité centrale de traitement permettant d'effectuer un accès au moyen de mémoire en vue d'une lecture et d'une écriture de donnée, si bien que, lorsque la donnée de transaction délivrée depuis un calculateur principal connecté au dispositif électronique portatif est emmagasinée dans le moyen de mémoire, l'unité centrale de traitement adjoint une donnée de détection d'erreur à la donnée de transaction, puis l'emmagasine.

La description suivante, conçue à titre d'illustration de l'invention, vise à donner une meilleure compréhension de ses caractéristiques et avantages; elle s'appuie sur les dessins annexés, parmi lesquels :

- la figure 1 est un schéma de principe montrant la disposition d'un système pour carte à circuit intégré selon un mode de réalisation de l'invention;
- la figure 2 est un schéma de principe fonctionnel d'une carte à circuit intégré selon l'invention;
- la figure 3 est un schéma de principe simplifié montrant un montage de circuit de la carte à circuit intégré selon l'invention;
- la figure 4 est une vue simplifiée montrant une mémoire de données divisée en plusieurs zones;
- la figure 5 est une vue plus détaillée d'une table d'index inscrite dans la zone 00 de la figure 4;
- la figure 6 est la représentation, ou format, des bits d'une donnée d'attribut inscrite en même temps que la donnée;
- la figure 7 est le format d'une donnée d'instruction d'écriture;
- la figure 8 est le format d'une donnée d'instruction de lecture;
- les figures 9A et 9B sont des organigrammes permettant d'expliquer une opération d'écriture de donnée pour la carte à circuit intégré selon l'invention;
- les figures 10A à 10E montrent les formats d'une donnée d'instruction et d'une donnée d'emmagasinage, tandis que la carte à circuit intégré selon l'invention effectue une opération d'écriture de donnée en réponse au format de donnée d'instruction d'écriture indiqué sur la figure 7;
- la figure 11 est une vue simplifiée montrant un état pour lequel une donnée est écrite dans une zone d'une mémoire; et
- les figures 12A et 12B sont des organigrammes servant à expliquer une opération de lecture de donnée qui est effectuée par la carte à circuit intégré selon l'invention, en réponse à l'instruction de lecture indiquée sur la figure 8.

Sur la figure 1, est présenté le montage d'un dispositif de manipulation de carte utilisé comme dispositif terminal dans un système de traitement bancaire ou un système d'achat à domicile, auquel une carte à circuit intégré, par exemple un dispositif électronique portatif selon un mode de réalisation de l'invention, est appliquée. Le dispositif de manipulation de carte possède un lecteur-enregistreur de carte 2 servant à accéder pour lecture ou écriture à une carte à circuit intégré, un clavier 4 servant à introduire des données, une unité 5 d'affichage du type tube cathodique servant à afficher des données introduites au clavier 4 et des données lues sur la carte à circuit intégré, une unité 7 de disque souple servant à emmagasiner les données, une imprimante 6 servant à imprimer les données, et une section de commande (CPU) 3 servant à commander le lecteur-enregistreur 2 de carte, le clavier 4, l'unité d'affichage 5, l'imprimante 6 et l'unité de disque souple 7.

La carte à circuit intégré détenue par l'utilisateur vérifie un mot de passe connu du seul utilisateur et emmagasine les données nécessaires lorsque l'utilisateur achète une marchandise. La carte à circuit intégré comprend une section de lecture-écriture 11, une section de fixation et, ou bien, de recueil de mot de passe 12, une section 13 de chiffrage-déchiffrage, et un superviseur 14 destiné à gérer ces sections, comme représenté sur la figure 2. La section de lecture-écriture 11 effectue la lecture des données, l'écriture des données ou leur effacement, par l'intermédiaire du lecteur-enregistreur de carte 2. La section de fixation-recueil de mot de passe 12 emmagasine un mot de passe fixé par l'utilisateur et effectue un traitement d'empêchement de lecture. En outre, la section 12 recueille le mot de passe, afin de permettre l'exécution d'un traitement ultérieur 3 après que le mot de passe a été fixé. La section de chiffrage-déchiffrage 13 chiffre et déchiffre les données, pour empêcher que des données de transmission ne soient décrites ou contrefaites pendant qu'elles sont émises en provenance du CPU 3 à destination d'un autre dispositif terminal via une ligne de transmission. Dans ce dispositif, le traitement de données s'exécute suivant un algorithme de chiffrage appartenant au Standard de chiffrage (Data Encryption Standard). Le superviseur 14 décode un code de

fonction ou bien un code de fonction avec donnée introduit en provenance du lecteur-enregistreur 2 de carte, et sélectionne et exécute toute fonction nécessaire.

Pour effectuer ces fonctions, comme représenté sur la figure 3, la carte à circuit intégré 1 comprend un élément de commande (par exemple un CPU) 15, une mémoire de données permanente 16 dont on peut effacer le contenu emmagasiné, une mémoire de programme 17, et un contacteur 18 servant à réaliser un contact électrique avec le lecteur-enregistreur de carte 2. Le CPU 15, la mémoire 16 et la mémoire de programme 17, qui sont entourés par une ligne en trait interrompu, sont constitués d'une puce de circuit intégré. La mémoire de programme 17 comprend une mémoire morte, ou ROM, à masque et emmagasine un programme servant à exécuter les fonctions ci-dessus mentionnées. La mémoire de données 16 comprend une mémoire morte programmable électriquement effaçable, ou EEPROM, et emmagasine diverses données de transaction.

Comme représenté sur la figure 4, la mémoire de données 16 est divisée en plusieurs zones. Chaque zone est constituée d'un ou plusieurs blocs. Chaque bloc est constitué par un nombre prédéterminé de bytes pour chaque zone et le traitement s'exécute en unités de blocs. Chaque bloc comprend une donnée d'attribut et une donnée d'emmagasinage. Les différentes zones sont numérotées de "00" à "FF". Comme le montre la figure 5, la zone "00" emmagasine le nombre de bytes de données d'emmagasinage se trouvant dans chaque bloc, dans les zones "01" à "FF", et les adresses de début et de fin des zones en correspondance l'une avec l'autre. En consultant la table d'index, on peut déterminer que l'adresse de début de la zone "01" est "aaa", son adresse finale est "bbb", et que le nombre de bytes par bloc est cinq bytes. Les parties hachurées de la figure 4 désignent des données d'attribut. La donnée d'attribut s'ajoute à chaque bloc et possède un identificateur indiquant si la donnée d'emmagasinage correspondante est effective ou non, et un identificateur indiquant si le bloc comporte la donnée finale lorsqu'une série de données d'emmagasinage a été emmagasinée en plusieurs blocs. La figure 6 montre le format de la donnée d'attribut. Le bit "6" est un identificateur signalant si la donnée d'emmagasinage se trouvant dans le

bloc est effective ou non. Si ce bit est "1", ceci indique que la donnée d'emmagasinage n'est pas effective et, si ce bit est "0", ceci indique que la donnée est effective. Le bit "7" est un identificateur indiquant si un bloc comporte ou non le byte final d'une série de données. Si ce bit est "1", ceci indique que le byte final n'est pas inclus dans ce bloc et, s'il s'agit de "0", ceci indique que le byte final y est contenu. On note que les bits "0" à "5" sont des bits fictifs.

Lorsque l'on inscrit une donnée dans la mémoire de données 16, on utilise une donnée d'instruction d'écriture, telle que représentée sur la figure 7. La donnée d'instruction d'écriture est constituée d'un code de fonction d'écriture, d'un numéro de zone et d'une donnée d'emmagasinage. Lorsqu'on lit une donnée dans la mémoire de données 16, on utilise la donnée d'instruction de lecture représentée sur la figure 8. La donnée d'instruction de lecture est constituée d'un code de fonction de lecture et d'un numéro de zone.

On va maintenant décrire, en relation avec les organigrammes des figures 9A et 9B, l'opération d'écriture de donnée, suivant l'agencement ci-dessus présenté, dans la mémoire de données 16.

Dans un état normal, le CPU 15 attend une donnée d'instruction d'écriture de la part du lecteur-enregistreur de carte 2. S'il est déterminé que la donnée d'instruction d'écriture a été introduite, le CPU 15 vérifie, au cours de l'étape 31, si le code de fonction inclus dans la donnée d'instruction est un code de fonction d'écriture. Si la réponse est OUI à l'étape 31, le CPU 15 consulte la table d'index, indiquée sur la figure 5, en utilisant un numéro de zone incorporé dans la donnée d'instruction comme paramètre, au cours de l'étape 33. Si la zone correspondante n'est pas trouvée, au cours de l'étape 33, le CPU 15 délivre une donnée de réponse indiquant que la zone n'est pas certifiée, au cours de l'étape 35, et le programme revient à l'étape 31. Toutefois, si la réponse est OUI à l'étape 31, c'est-à-dire si la zone correspondante a été trouvée, le CPU 15 emmagasine la donnée de l'unité de traitement, et les adresses de début et de fin correspondant au numéro de zone, dans la RAM 15a. Le CPU 15 produit un code de contrôle pour la donnée d'emmagasinage incluse dans la donnée d'instruction, et l'adjoint

à l'extrémité de la donnée d'emmagasinage. Le nombre de bytes de la donnée d'emmagasinage contenant le code de contrôle est compté et est temporairement emmagasiné dans la RAM 15a. Il est possible de produire le code de contrôle en créant un code de somme de contrôle pour la donnée de bytes respective de la donnée d'emmagasinage ou en appliquant une fonction OU exclusif à la donnée de bytes respective. Lorsque la donnée d'emmagasinage associée au code de contrôle, qui est temporairement emmagasinée dans la RAM 15a, a été écrite dans une zone spécifiée par la donnée d'instruction, le CPU 15 recherche l'adresse de début d'une zone vierge, au cours de l'étape 37. Si la réponse est NON au cours de l'étape 37, le CPU 15 délivre une donnée de réponse indiquant qu'il n'existe pas de zones vierges, au cours de l'étape 39, et le programme revient à l'étape 31. Toutefois, si la réponse est OUI à l'étape 37, l'adresse de début est temporairement emmagasinée dans la RAM 15a. Le CPU 15 adjoint le nombre de bytes emmagasiné de la donnée d'emmagasinage à la tête de la donnée d'emmagasinage, et la subdivise en unités de traitement, au cours de l'étape 41. Le CPU 15 écrit la première donnée subdivisée dans la zone spécifiée, au cours de l'étape 43. Le CPU 15 contrôle, au cours de l'étape 45, si la donnée a été correctement écrite dans la zone. Ceci peut être exécuté par exemple par une lecture immédiate de la donnée après son écriture, puis comparé avec la donnée d'entrée initiale. Si la réponse est NON au cours de l'étape 45, le CPU 15 adjoint une donnée d'attribut indiquant que la donnée subdivisée n'est pas effective, au cours de l'étape 47. Au cours de l'étape 49, le CPU 15 écrit de nouveau la donnée subdivisée dans la zone vierge suivante, puis le programme revient à l'étape 45. Toutefois, si la réponse est OUI à l'étape 45, le CPU 15 adjoint une donnée d'attribut indiquant que la donnée subdivisée est effective, au cours de l'étape 51. Le CPU 15 contrôle ensuite, au cours de l'étape 53, si toute la donnée d'emmagasinage a bien été écrite dans la zone spécifiée. Si la réponse est NON au cours de l'étape 53, le CPU 15 écrit la donnée subdivisée suivante dans la zone vierge suivante, au cours de l'étape 55, puis le programme revient à l'étape 45.

Toutefois, si la réponse est OUI à l'étape 53, le CPU 15 délivre une donnée de réponse indiquant l'achèvement de

l'opération d'écriture, au cours de l'étape 57, puis le programme revient à l'étape 31. Ensuite, le CPU 15 attend la donnée d'instruction suivante. Dans le mode d'écriture, dans un bloc dans lequel le byte final de la donnée d'emmagasinage a été écrit, le bit "7" de la donnée d'attribut de ce bloc est positionné à "0".

On suppose que la donnée d'instruction d'écriture présentée sur la figure 10A a été introduite. Dans ce cas, puisque le numéro de zone est "02", le nombre de bytes de l'unité à traiter est quatre, comme on peut le voir dans la table d'index de la figure 5. Un code d'instruction d'écriture est certifié, et il est également certifié qu'il existe des zones vierges. Ensuite, la donnée d'emmagasinage contenue dans la donnée d'instruction d'entrée est extraite (voir figure 10B). Ensuite, un code de contrôle est adjoint à la donnée d'emmagasinage, comme représenté sur la figure 10C, et la longueur de la donnée d'emmagasinage est inscrite immédiatement avant la tête de la donnée d'emmagasinage, comme représenté sur la figure 10D. La donnée d'emmagasinage, qui est précédée par la longueur de la donnée, est subdivisée en bytes de l'unité en vue du traitement, comme représenté sur la figure 10E, et est emmagasinagée dans la zone "02", comme représenté sur la figure 11. Comme le montre la figure 11, les parties hachurées des bytes "1", "6", "11" et "16" indiquent la donnée d'attribut. Dans ce cas, un drapeau indiquant le bloc final est positionné dans la donnée d'attribut, à savoir dans le byte "16". La donnée d'attribut des bytes "1", "11" et "16" indique que la donnée est effective.

On va maintenant décrire, en relation avec les organigrammes des figures 12A et 12B, le processus de lecture d'une donnée dans la mémoire 16. Dans l'état normal, la carte à circuit intégré attend une donnée d'instruction de lecture de la part du lecteur-enregistreur de carte 2. Si une donnée d'instruction est introduite depuis le lecteur-enregistreur de carte 2, le CPU 15 vérifie d'abord, au cours de l'étape 59, si le code de fonction inclus dans la donnée d'instruction est un code de fonction de lecture. Si la réponse est OUI à l'étape 59, le CPU 15 recherche le numéro de zone adjoint à la donnée d'instruction dans la mémoire de données 16 à partir de la zone "00", au cours de l'étape 61. Si la réponse est NON à

l'étape 61, le CPU 15 délivre une donnée de réponse indiquant que la zone n'est pas certifiée, à l'étape 63, puis le programme revient à l'étape 59. Toutefois, si la réponse est OUI à l'étape 61, le CPU 15 emmagasine le nombre correspondant de bytes de l'unité en vue du traitement, et les adresses de début et de fin de la zone, dans la RAM 15a. A l'étape 65, le CPU 15 recherche un bloc qui comporte la donnée de début devant être lue. Une fois un tel bloc trouvé, à l'étape 65, la longueur de donnée précédant la tête de ce bloc est positionnée dans un compteur programmé qui est emmagasiné dans la RAM 15a, comme valeur initiale, au cours de l'étape 67. A l'étape 69, le CPU 15 lit le byte de donnée suivant et vérifie, au cours de l'étape 71, si la donnée lue est une donnée d'attribut. Si la réponse est OUI à l'étape 71, le programme revient à l'étape 69, et le CPU 15 lit le byte suivant. Si la réponse est NON à l'étape 71, le CPU 15 emmagasine la donnée lue dans la RAM 15a, au cours de l'étape 73. Au cours de l'étape 75, le CPU 15 décrémente d'une unité le compteur programmé. Le CPU 15 vérifie, au cours de l'étape 77, si le compteur a atteint "0". Si la réponse est NON à l'étape 77, le programme revient à l'étape 69, et les étapes 69 à 77 se répètent. De cette manière, si la réponse OUI est obtenue à l'étape 77, toute la chaîne de données introduite dans le mode d'écriture est temporairement emmagasinée dans la RAM 15a, sans lecture des données fictives. Au cours de l'étape 79, la chaîne de données temporairement emmagasinée subit une modification. Plus spécialement, des sommes de contrôle pour la donnée emmagasinée dans la RAM 15a sont calculées pour chaque byte, ou bien les bytes respectifs de la donnée sont soumis à une fonction OU exclusif, puis la donnée résultante est comparée avec le code de contrôle écrit dans le byte final de la chaîne de données. Si le CPU 15 décide, au cours de l'étape 81, que la donnée n'est pas valable, il délivre une donnée de réponse, indiquant que la donnée n'est pas valable, au lecteur-enregistreur 2 de carte à circuit intégré, au cours de l'étape 83, puis le programme revient à l'étape 59. Toutefois, s'il est décidé, au cours de l'étape 81, que la donnée est valable, le CPU 15 délivre la chaîne de données qui est emmagasinée dans la RAM 15a au lecteur-enregistreur 2 de carte à circuit intégré, au cours de l'étape 85.

Dans ce mode de réalisation, un code de contrôle adjoint à la donnée d'emmagasinage est produit dans le CPU 15. Toutefois, un code de contrôle introduit depuis un dispositif externe, par exemple le lecteur-enregistreur 2 de carte, peut être adjoint à la donnée d'emmagasinage et être emmagasiné.

Dans ce mode de réalisation, on a choisi comme exemple de dispositif électronique portatif une carte à circuit intégré. La forme du dispositif électronique portatif n'est pas limitée à la forme d'une carte, mais peut être celle d'un bloc ou d'un crayon.

Bien entendu, l'homme de l'art sera en mesure d'imaginer, à partir du dispositif dont la description vient d'être donnée à titre simplement illustratif et nullement limitatif, diverses variantes et modifications ne sortant pas du cadre de l'invention.

REVENDICATIONS

1. Dispositif électronique portatif comportant un moyen de mémoire de données (15a) servant à emmagasiner des données de transaction; et une unité centrale de traitement (15) servant à exécuter une opération d'accès en lecture et écriture de donnée dans ledit moyen de mémoire de données (15a), caractérisé en ce que, lorsque la donnée de transaction, qui est délivrée depuis un système principal connecté audit dispositif électronique portatif, est emmagasinée dans ledit moyen de mémoire de données (15a), ladite unité centrale de traitement adjoint une donnée de détection d'erreur à la donnée de transaction, puis l'emmagasine.

2. Dispositif selon la revendication 1, caractérisé en ce que ladite unité centrale de traitement (15) produit la donnée de détection d'erreur en créant un code de somme de contrôle pour la donnée de transaction.

3. Dispositif selon la revendication 1, caractérisé en ce que ladite unité centrale de traitement (15) produit la donnée de détection d'erreur en créant un code à traitement par une fonction OU exclusif pour la donnée de transaction.

4. Dispositif selon la revendication 2 ou 3, caractérisé en ce que l'unité centrale de traitement (15) compte la longueur de la donnée de la transaction contenant le code de somme de contrôle ou le code ayant subi une fonction OU exclusif, et emmagasine le code de somme de contrôle ou le code ayant subi une fonction OU exclusif, comme donnée de détection d'erreur, dans le moyen de mémoire de données (15a) en même temps que la donnée de transaction.

5. Dispositif selon la revendication 4, caractérisé en ce que, lors de la lecture de la donnée de transaction emmagasinée dans le moyen de mémoire de données (15a), l'unité centrale de traitement (15) produit le code de somme de contrôle ou le code ayant subi une fonction OU exclusif, compte la longueur de la donnée de transaction comportant le code de somme de contrôle ou le code ayant subi la fonction OU exclusif, compare le code de somme de contrôle ou le code ayant subi une fonction OU exclusif produits et la longueur de

donnée comptée, avec le code de somme de contrôle ou le code ayant subi une fonction OU exclusif et la longueur de la chaîne de données emmagasinés dans ledit moyen de mémoire de données (15a), et décide que la donnée de transaction écrite est valable lorsqu'une coïncidence a été trouvée.

6. Dispositif selon la revendication 1, caractérisé en ce que ledit moyen de mémoire de données (15a) est constitué d'une mémoire permanente.

FIG. 1

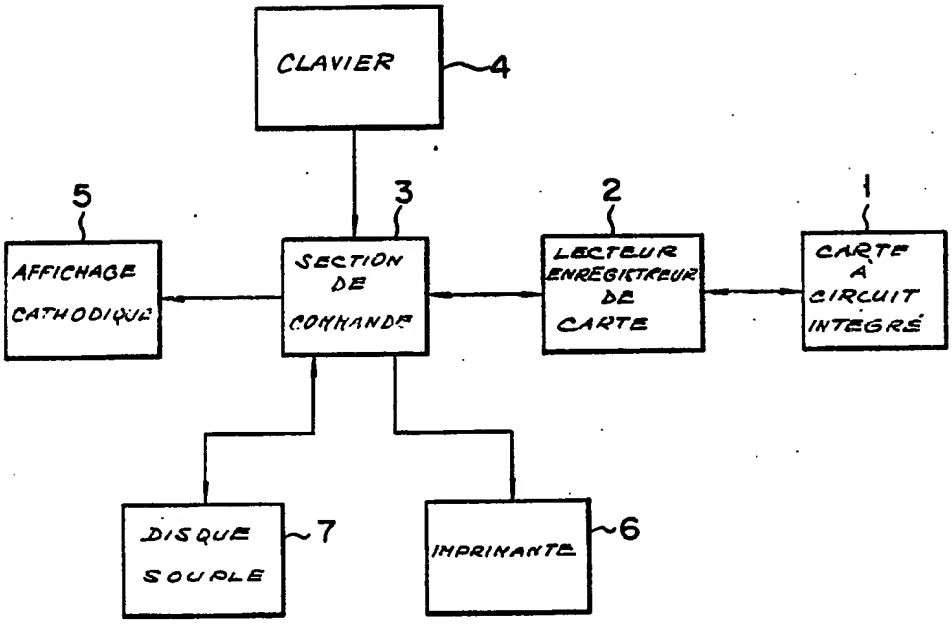

CLAVIER
4
5
3
2
1
AFFICHAGE CATHODIQUE
SECTION DE COMMANDE
LECTEUR ENREGISTREUR DE CARTE
CARTE A CIRCUIT INTEGRÉ
DISQUE SOUPLE
7
IMPRIMANTE
6

FIG. 2

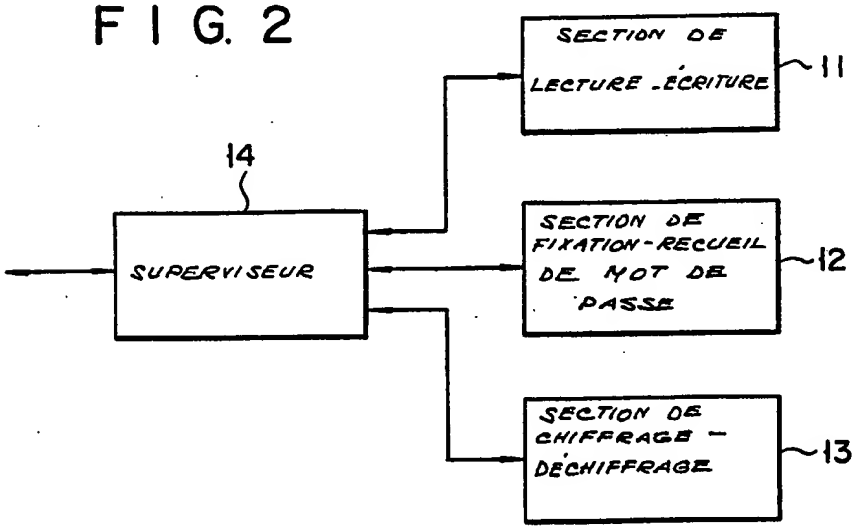

SECTION DE LECTURE - ÉCRITURE
11
14
SUPERVISEUR
SECTION DE FIXATION - RECUEIL DE MOT DE PASSE
12
SECTION DE CHIFFRAGE - DÉCHIFFRAGE
13

FIG. 3

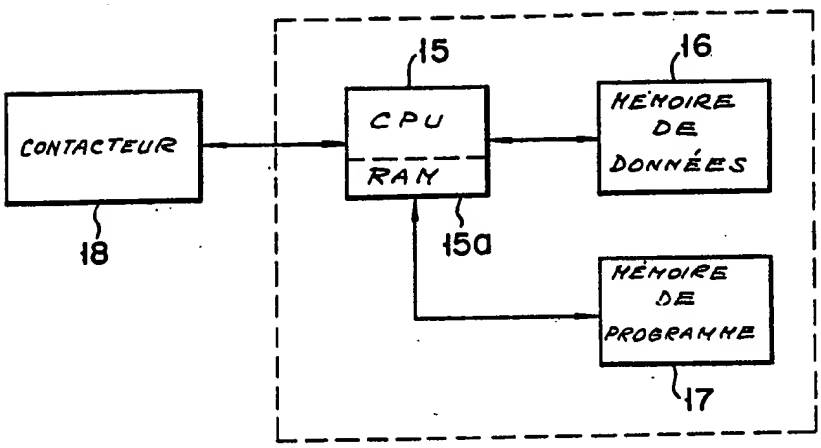

15
16
CPU
RAM
MÉMOIRE DE DONNÉES
CONTACTEUR
18
15a
MÉMOIRE DE PROGRAMME
17

FIG. 4

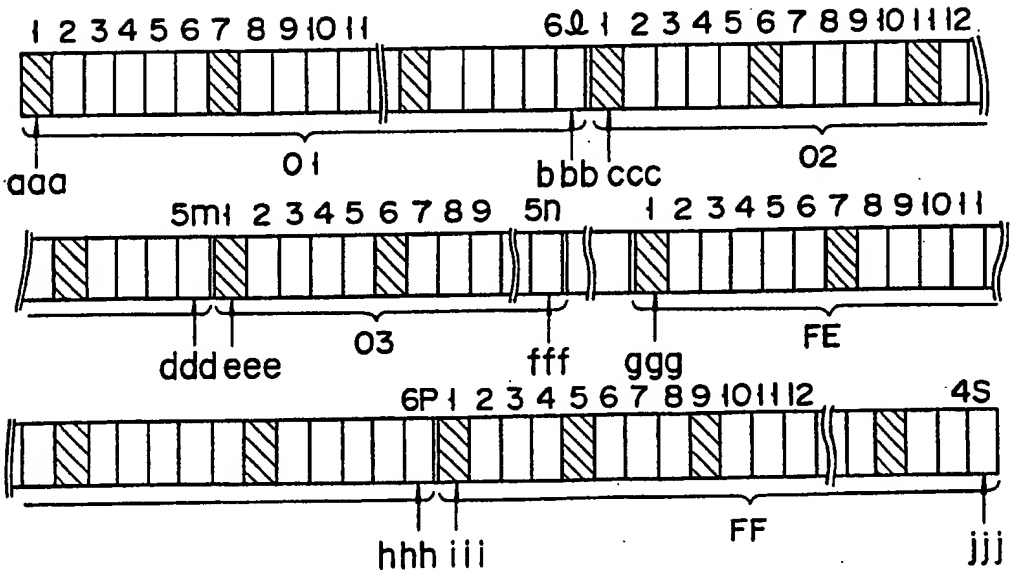

1 2 3 4 5 6 7 8 9 10 11
6Q 1 2 3 4 5 6 7 8 9 10 11 12
aaa
01
bbb ccc
02
5m 1 2 3 4 5 6 7 8 9
5n
1 2 3 4 5 6 7 8 9 10 11
ddd eee
03
fff
ggg
FE
6P 1 2 3 4 5 6 7 8 9 10 11 12
4S
hhh iii
FF
jjj

FIG. 5

NUMERO DE ZONE	NOMBRE DE BYTES	ADRESSE DE DÉBUT	ADRESSE DE FIN
01	5	aaa	bbb
02	4	ccc	ddd
03	4	eee	fff
FE	5	ggg	hhh
FF	3	iii	jjj

FIG. 6

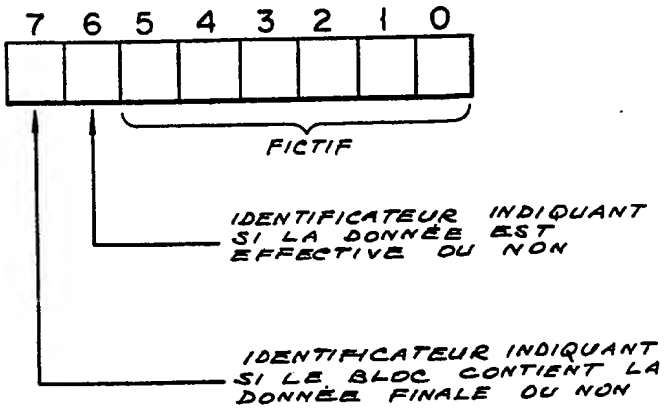


FIG. 7

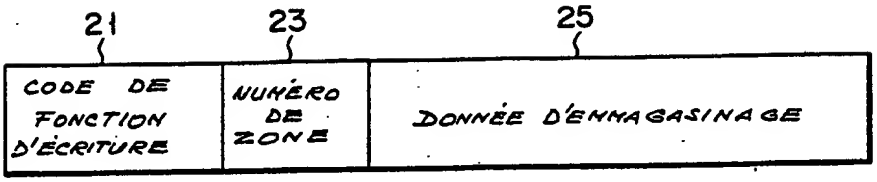


FIG. 8

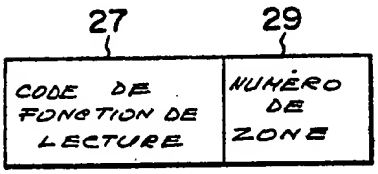


FIG. 11

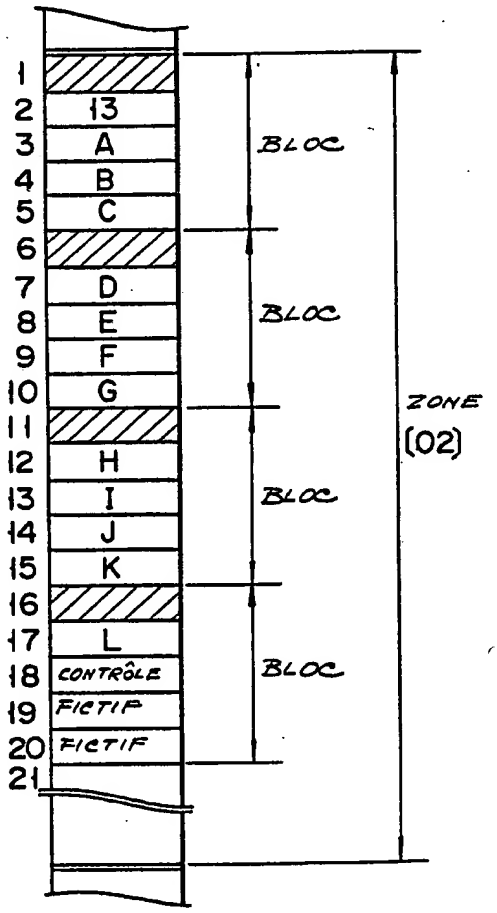


FIG. 9A

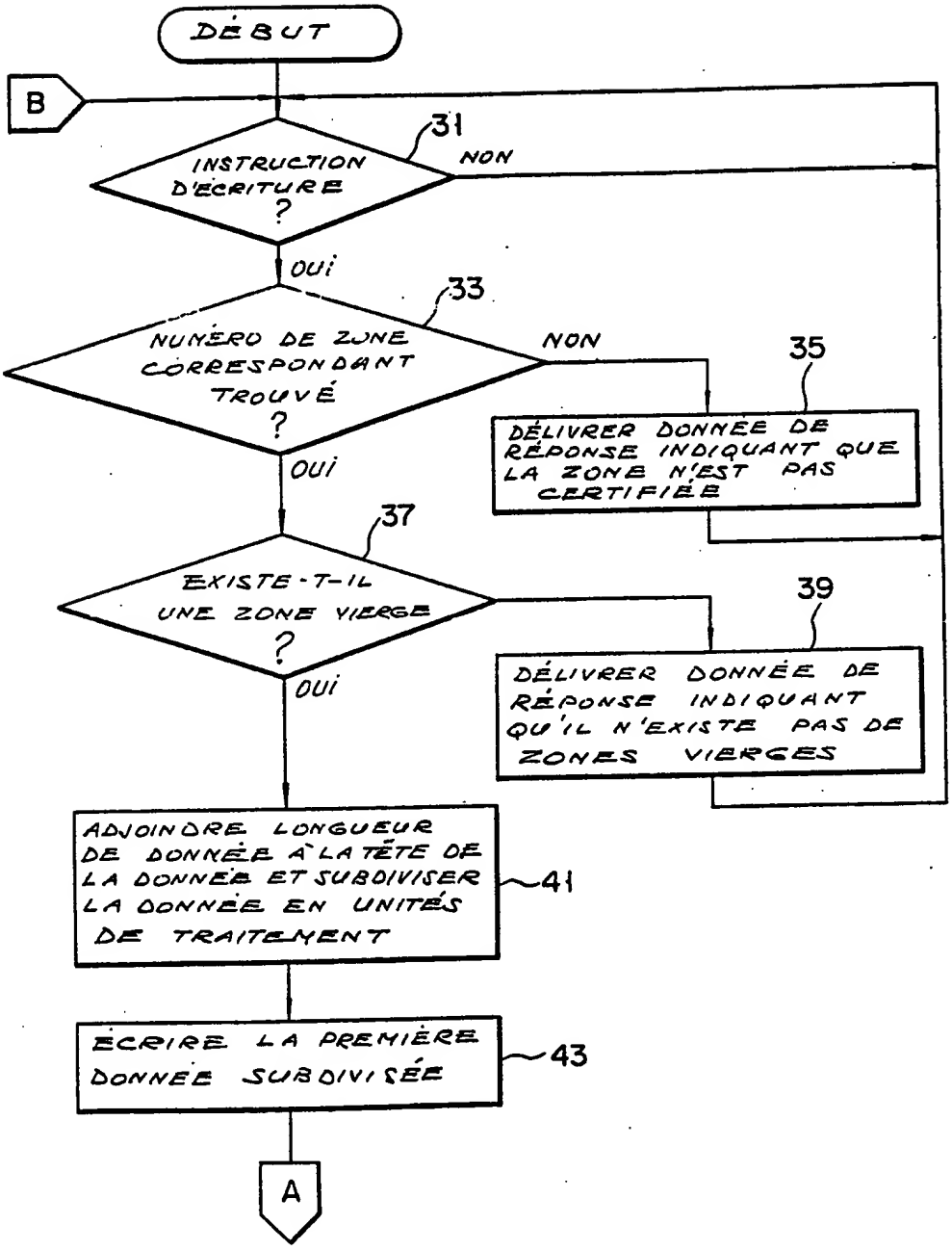

DÉBUT
B
INSTRUCTION D'ÉCRITURE ?
31
NON
OUI
NUMÉRO DE ZONE CORRESPONDANT TROUVÉ ?
33
NON
35
DÉLIVRER DONNÉE DE RÉPONSE INDIQUANT QUE LA ZONE N'EST PAS CERTIFIÉE
OUI
EXISTE-T-IL UNE ZONE VIERGE ?
37
39
DÉLIVRER DONNÉE DE RÉPONSE INDIQUANT QU'IL N'EXISTE PAS DE ZONES VIERGES
OUI
AJOINDRE LONGUEUR DE DONNÉE À LA TÊTE DE LA DONNÉE ET SUBDIVISER LA DONNÉE EN UNITÉS DE TRAITEMENT
41
ÉCRIRE LA PREMIÈRE DONNÉE SUBDIVISÉE
43
A

FIG. 9B

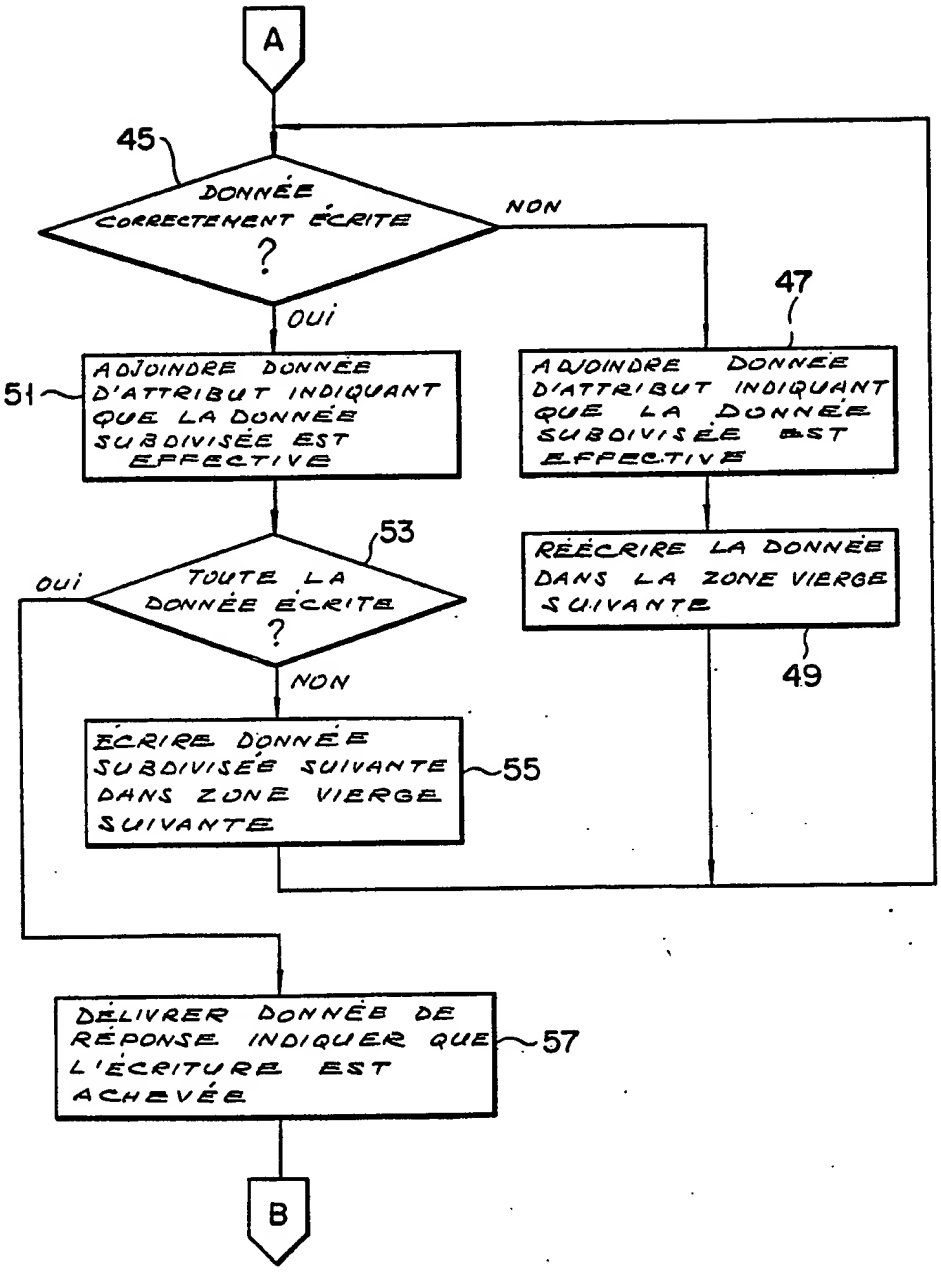

A
45
DONNÉE CORRECTEMENT ÉCRITE ?
NON
OUI
47
AJOINDRE DONNÉE D'ATTRIBUT INDIQUANT QUE LA DONNÉE SUBDIVISÉE EST EFFECTIVE
51
AJOINDRE DONNÉE D'ATTRIBUT INDIQUANT QUE LA DONNÉE SUBDIVISÉE EST EFFECTIVE
RÉÉCRIRE LA DONNÉE DANS LA ZONE VIERGE SUIVANTE
49
53
OUI
TOUTE LA DONNÉE ÉCRITE ?
NON
ÉCRIRE DONNÉE SUBDIVISÉE SUIVANTE DANS ZONE VIERGE SUIVANTE
55
DÉLIVRER DONNÉE DE RÉPONSE INDIQUER QUE L'ÉCRITURE EST ACHEVÉE
57
B

FIG. 10A

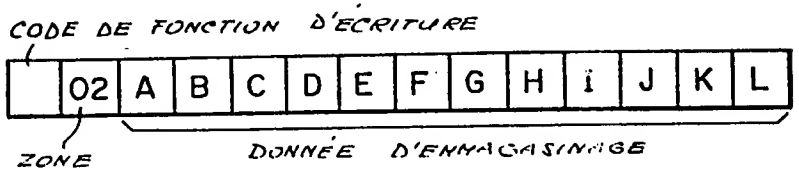


FIG. 10B

FIG. 10C

FIG. 10D

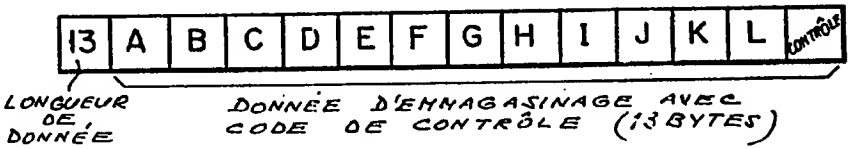
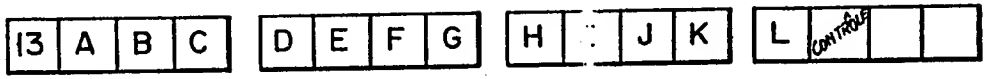


FIG. 10E



2594573

FIG. 12A

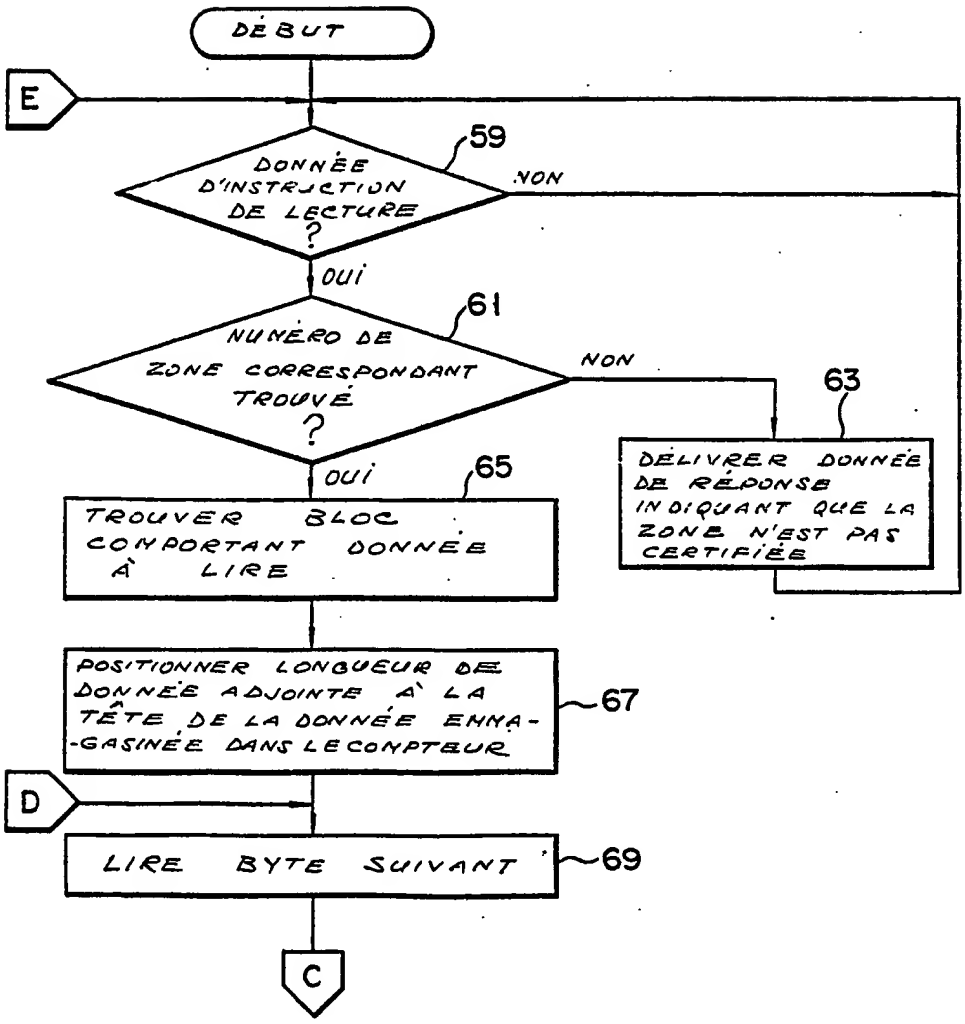
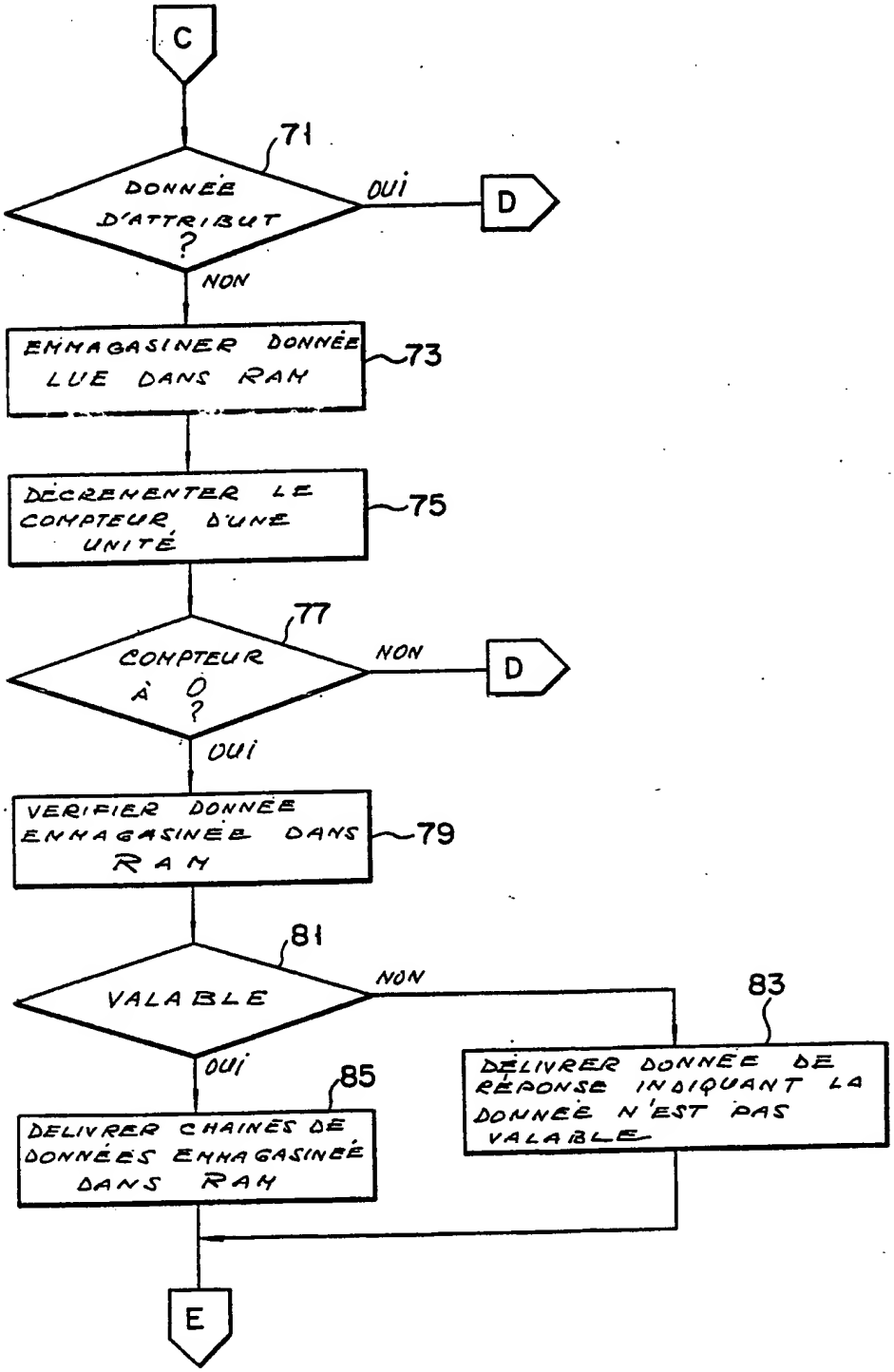

DÉBUT
E
59
DONNÉE D'INSTRUCTION DE LECTURE ?
NON
OUI
61
NUMÉRO DE ZONE CORRESPONDANT TROUVÉ ?
NON
63
DÉLIVRER DONNÉE DE RÉPONSE INDIQUANT QUE LA ZONE N'EST PAS CERTIFIÉE
OUI
65
TROUVER BLOC COMPORTANT DONNÉE À LIRE
POSITIONNER LONGUEUR DE DONNÉE ADJOINTE À LA TÊTE DE LA DONNÉE EMMAGASINÉE DANS LE COMPTEUR
67
D
LIRE BYTE SUIVANT
69
C

FIG. 12B


C
71
DONNÉE D'ATTRIBUT ?
OUI
D
NON
ENMAGASINER DONNÉE LUE DANS RAM
73
DÉCREMENTER LE COMPTEUR D'UNE UNITÉ
75
77
COMPTEUR À 0 ?
NON
D
OUI
VÉRIFIER DONNÉE ENMAGASINÉE DANS RAM
79
81
VALABLE
NON
83
DÉLIVRER DONNÉE DE RÉPONSE INDIQUANT LA DONNÉE N'EST PAS VALABLE
OUI
85
DÉLIVRER CHAINES DE DONNÉES ENMAGASINÉE DANS RAM
E